

Simulation of 4D printed hydrogel Using RNN

Yifan Xu¹, Mengtao Wang¹, Zhongkui Wang² and Lin Meng^{3,*}

¹Graduate School of Science and Engineering, Ritsumeikan University, 1-1-1 Noji-higashi, Kusatsu, Shiga, 525-8577, Japan

²College of Robotics, Ritsumeikan University, 1-1-1 Noji-higashi, Kusatsu, Shiga, 525-8577, Japan

³College of Science and Engineering, Ritsumeikan University, 1-1-1 Noji-higashi, Kusatsu, Shiga, 525-8577, Japan

Abstract

This study proposes a method to shorten the deformation simulation time of four-dimensionally printed hydrogel models using deep learning. In this method, a large number of hydrogel models with the same shape but different distribution of expansion ratios are first created using Abaqus, and then deformation simulations are performed using the Abaqus thermal expansion method, and a dataset is created based on the simulation results. A recurrent neural network (RNN) was created and trained on this dataset, allowing the RNN to learn model deformation features that can be used to predict the deformation of longer hydrogels under the same loading conditions. This research provides a method for efficient simulation and inverse modeling of 4D printing.

Keywords

4D printing, Deep Learning, Recurrent Neural Network

1. Introduction

In recent years, 3D printing technology has become an important technology in the manufacturing and engineering fields [1], bringing innovation to product design and production. However, as the technology develops, more advanced manufacturing technologies are attracting attention, one of which is 4D printing technology [2]. Compared to traditional 3D printing technology, 4D printing technology has unique advantages [3]. 4D printing technology can not only create objects in three-dimensional space, but also control changes in the fourth dimension, time. This temporal control allows the fabricated object to automatically adjust its shape, structure, or function in response to external stimuli, achieving a higher degree of adaptability and intelligence[4]. Smart materials such as hydrogels are used in 4D printing technology. Hydrogels undergo reversible or irreversible deformation under stimuli or specific conditions [5]. With the use of smart materials, the fabricated products are equipped with a wide variety of functions such as self-healing, self-folding, and self-expanding, offering new application possibilities in various fields. 4D printing technology can achieve highly individualized

The 6th International Symposium on Advanced Technologies and Applications in the Internet of Things (ATAIT 2024), August 19-22, 2024, Kusatsu, Shiga, Japan


*Corresponding author.

✉ gr0623hx@ed.ritsumei.ac.jp (Y. Xu); gr0663sx@ed.ritsumei.ac.jp (M. Wang); wangzk@fc.ritsumei.ac.jp (Z. Wang); menglin@fc.ritsumei.ac.jp (L. Meng)

🆔 0009-0003-0035-7851 (Y. Xu); 0009-0003-9815-6838 (M. Wang); 0000-0003-4037-3306 (Z. Wang);

0000-0003-4351-6923 (L. Meng)

© 2024 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

manufacturing. It can customize products with different shapes and functions to meet specific needs and environments. Due to this adaptability, 4D printing technology has great potential in a variety of fields, including smart materials, bio-medicine, and wearable technology [6].

However, while there are many advantages to 4D printing technology, there are some challenges and limitations to its practical application. One of these problems is related to modeling and simulation. Simulating the deformation of long sequences of hydrogels requires a large number of calculations, which will consume a lot of time and computational resources [7]. This is a limiting factor for practical applications. In addition to this, due to the large number of physical parameters of the model with complex simulation settings, it is often difficult to obtain a globally optimal solution when optimizing the model parameters. To overcome these difficulties, we propose a new method. That is, the recurrent neural network (RNN) is utilized to predict the simulation results of the model based on the initial sequence of the hydrogel model [8]. This method can replace the traditional simulation process, avoid the consumption of time and computational resources caused by simulation, and improve the efficiency and feasibility of 4D printing technology [9]. In terms of model parameter optimization, since the method is based entirely on data-level computation and processing, it avoids the complex effects of the physical level, making it simpler to solve for the global optimal solution.

The contributions made by this study are as follows:

1. This study is the first to apply RNN models to a hydrogel modeling method based on 4D printing technology, and demonstrates the superiority in time and computational space.

2. The RNN model used in this study can be used not only for the deformation prediction of hydrogel models of the same length, but also for hydrogel models of different lengths with high accuracy. This advantage makes the technology not only suitable for hydrogel models, but also for the production of 4d printed models with modular structures.

2. Hydrogel Model Structure

The model for this study was created using software named Abaqus. As shown in Figure 1, the entire model consists of two identically shaped sections stacked above and below each other, with each section a rectangle 10 mm long, 2 mm wide, and 2 mm thick. Divide each layer into parts in millimeter increments in the longitudinal direction. The numbers "0" and "1" represent the different expansion rates of each part. "0" shows hydrogel with the expansion rate of 1.8938 and "1" shows the expansion of 2.1223. The Random Forest Model (RFM) generates rectangles arranged in various orders. Each part has exactly the same material properties such as density, Poisson's ratio, Young's modulus, specific heat, and thermal conductivity, except for the expansion coefficient.

Based on the above material properties, the hydrogel model will continuously absorb water and swell in water to produce deformation. The whole process of deformation can be clearly reflected according to the time change, while the deformation result depends on the initial swelling rate sequence of the model. It can be seen that the sequence and deformation of the hydrogel model are standard time series data.

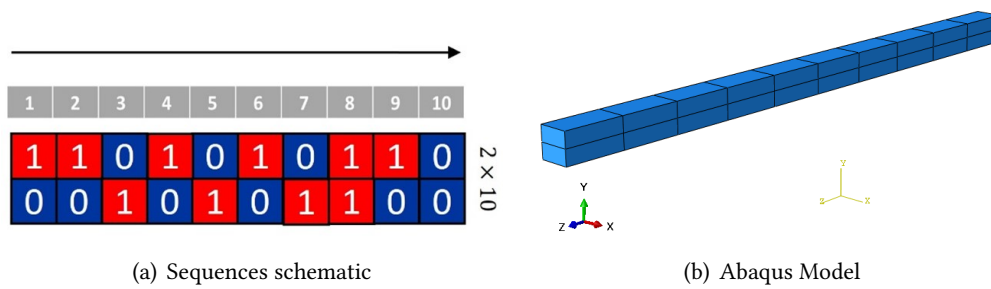


Figure 1: Model Structure:(a) shows the hydrogel model built in Abaqus software; (b) shows the schematic of one of the different swelling rate sequences of the hydrogel model

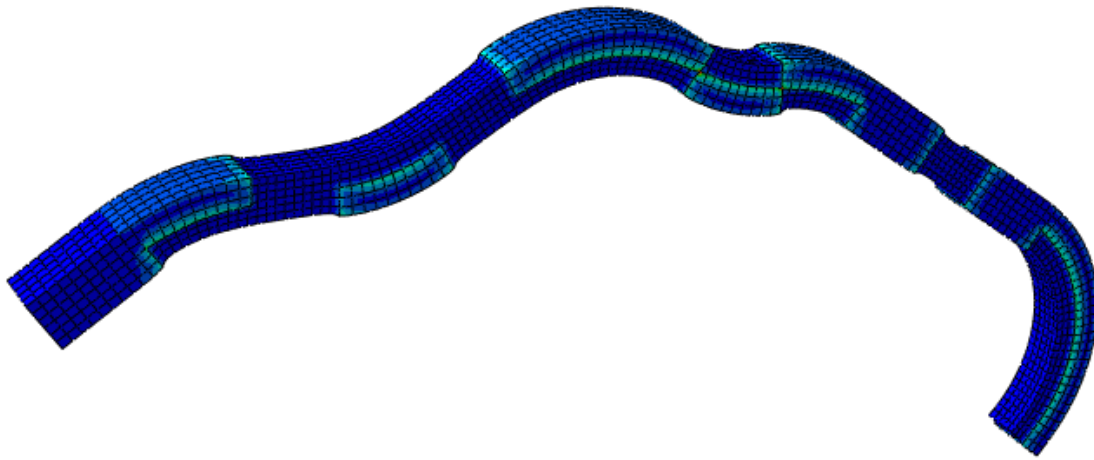


Figure 2: simulation result:The results of the simulation of the deformation of the hydrogel model after finite element analysis are shown, in which the leftmost lower edge is fixed and cannot move freely.

3. DataSet Creation

After placing the model in water, the model is deformed by expansion due to the higher water absorption of the hydrogel, and the different expansion rates of the parts cause the model to bend and deform into a curvilinear shape. To simulate the deformation process, a finite element analysis is performed using Abaqus. As an equivalent substitute for water absorption and expansion, the simulated loads were set to vary in temperature from 0°C to 15°C. 2 shows the result of the simulation. At the same time, one side of the model is fixed so that the deformation of the model occurs only in the x-axis and z-axis directions, which is convenient for the final result processing. For processing the simulation results, 20 reference points are selected at 1 mm intervals in the deformed model and the coordinates of each reference point are output. In total, 400 reference points will create a curve showing the simulation results. Models with different expansion rates will obtain different deformation curves after simulation. After simulating 1000 of these different models, 1000 different curves are obtained. The data set of the hydrogel model

| | | Origin Sequence | | | | | | | | | | Final Direction | | | | | | | | | |
|--------------------|-----|-----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----------------|-----|-----|------|------|-----|------|--------------------|--|--|
| | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | ... | 219 | | | |
| two-layer hydrogel | | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 2.5 | 3.5 | 3.7 | 4.2 | 4.5 | ... | 612 | → Z-axis direction | | |
| | | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 10 | 6.7 | 3.3 | -3.4 | -6.7 | ... | -280 | → X-axis direction | | |
| | | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 2.5 | 2.1 | 1.9 | 1.5 | 1.1 | ... | -605 | | | |
| | | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 10 | 5.5 | 1.1 | -3.2 | -7.6 | ... | -303 | | | |
| | | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 2.5 | 2.4 | 2.5 | 2.4 | 2.5 | ... | 524 | | | |
| | | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 10 | 6.6 | 3.3 | 0.0 | -3.3 | ... | -445 | | | |
| | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | | | |
| | | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 2.5 | 2.5 | 2.5 | 2.5 | 2.5 | ... | 167 | | | |
| | | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 10 | 6.6 | 3.3 | -0.1 | -3.3 | ... | -661 | | | |

Figure 3: The sequence structure of the two hydrogel layers and the x-axis and z-axis displacements of the datum are stored in a table

is obtained by storing all the coordinates of the obtained curve reference points corresponding to the "0" and "1" sequence codes of the initial model in the form of a table. As shown in the figure 3, each row represents one of the layers in the two-layer hydrogel model, and every two rows form a piece of the hydrogel model. The 10 columns on the left side of the table encode the hydrogel "0" and "1" sequence before the deformation, while the 200 columns on the right side correspond to the coordinates of the reference points after the deformation. The first and second rows of each group represent the coordinates of the reference point in the x-axis direction and in the z-axis direction, respectively.

4. RNN Structure

Recurrent Neural Network (RNN) is a neural network characterized by its ability to take into account not only the forward flow of information but also the backward flow of information. That is, inputs corresponding to the present point in time not only produce outputs for the present point in time, but also outputs the effect on the subsequent point in time, which makes it possible to react the sequence to the global effect. This property matches the material properties of the hydrogel model, so this study chose to use RNN as the model for deep learning [10]. In addition, a special RNN derivative called LSTM (Long Short-Term Memory) has the ability to learn long-term dependencies and can effectively understand long-term relationships in time series data [11]. Due to the advantage of flexibility and versatility, RNNs can be adapted to a wide variety of problems, and in this study, RNNs are used to predict deformations [12]. As Figure 4 shows, the RNN employed in this experiment alternates between four LSTM layers with 64 neurons and four tightly coupled layers with 64 neurons.

Overall, the alternate use of the LSTM layer and the fully-connected layer enables the RNN model used in this experiment to capture the long-term dependencies in the time-series data and to perform feature extraction and dimensionality reduction on the captured information, which leads to a better representation of the states and features of the hydrogel model and a more accurate prediction of the future state of the hydrogel model, and has yielded good results

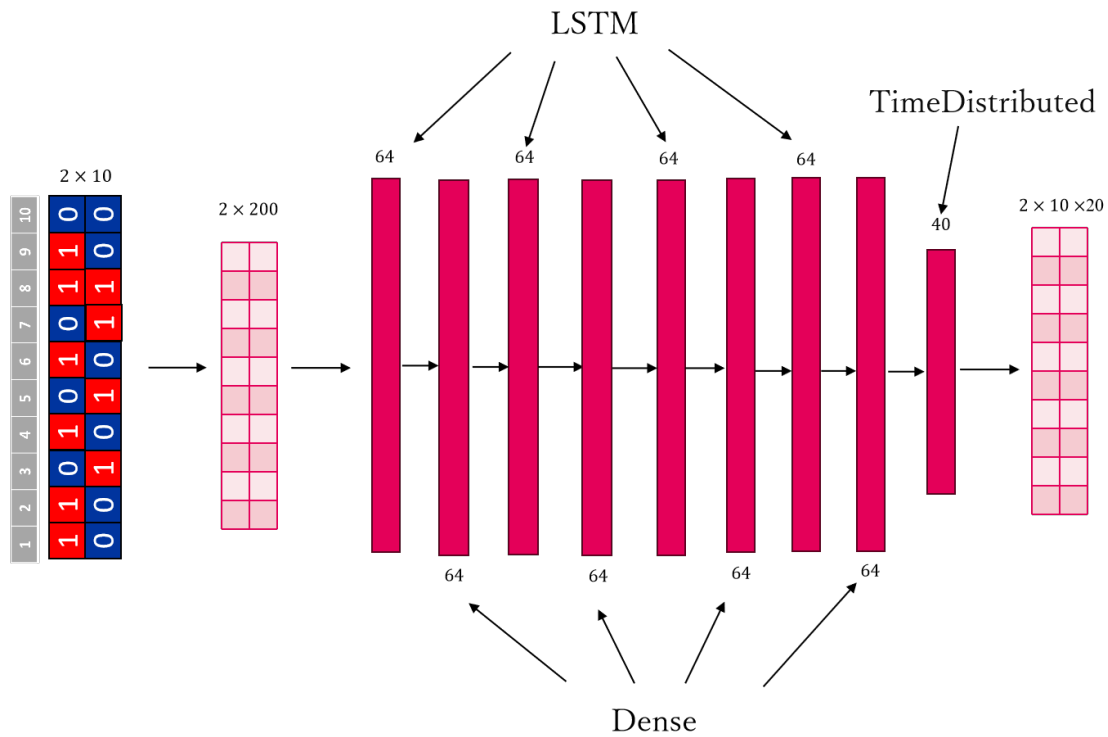


Figure 4: RNN structure: The data of 2x10 structure is converted to 2x200 and then sequentially fed into the neural network consisting of the intersection of LSTM and Dense layers, and finally output as 2x10x20 data structure.

in the experimental results [13].

4.1. LSTM Structure

As the figure 5(a) shows, the LSTM layer, as a kind of recurrent neural network, has good memory capability, it can effectively capture the long-term dependencies in time series data [14]. In this experiment, each LSTM layer is designed to learn the state changes of the hydrogel model at different time steps and optimize the model parameters to predict the future state more accurately through the back propagation process of the network.

4.2. Dense Structure

Second, for the four tightly packed dense layers with 64 neurons employed. As the Figure 5(b) shows, These dense layers are designed to take the output of the LSTM layer for downscaling and feature extraction to better represent the states and features of the hydrogel model. During the training process, these fully connected layers help the network better understand the structure and changing patterns of the hydrogel model by learning the relationship between the inputs and outputs of the hydrogel model.

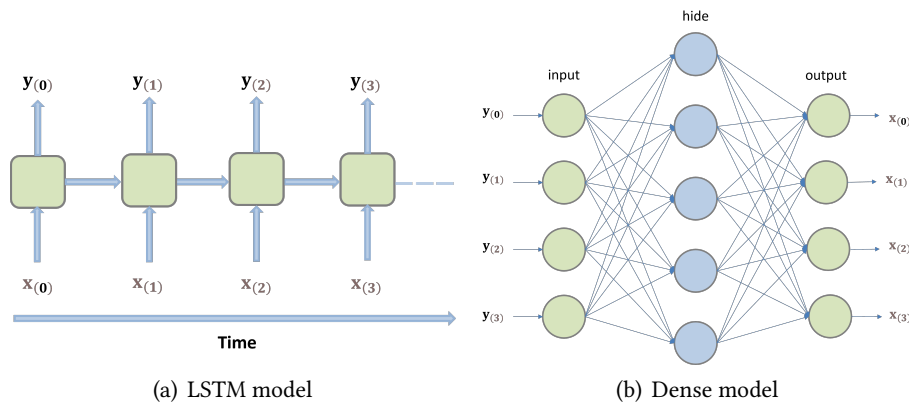


Figure 5: Model structure: The data is partitioned into $x(i)$ sequences by time step order into the LSTM layer, the output $y(i)$ sequences as the input to the Dense layer through the Dense layer to process the data, the results of the processing is done as the input to the next LSTM layer, the cycle of the process to complete the extraction of features.

5. Prediction Process for Deep Learning

5.1. Training Process

For the training part, as the figure 6 shows, the parameters of the RNN model including weights and biases are initialized first, and the above parameters will be updated gradually during the training process so that the model can fit the training data better. After the initialization of the model is completed, the short sequence data with a length of 10 cm obtained from the finite element analysis is used as the training data, which contains different sequence alignments as input data and the deformed coordinates as predicted target data. After inputting the training data into the RNN model to start training, the loss function is utilized to calculate the difference between the predicted results of the model and the real results each time, which is called loss. The loss results in each training session are used as a measure of model accuracy, and the backpropagation process uses an optimization algorithm to update the parameters of the model so that the model fits the training data better. After repeating the steps of forward propagation to calculate the loss and backpropagation to update the parameters until the predefined convergence threshold of the loss function is reached, the RNN model will learn the pattern of change of the hydrogel model and try to predict the future state at each time step in the subsequent process [15].

5.2. Prediction Process

This is followed by the prediction part. As the figure 7 shows, where a short sequence of length 10cm is first used as the initial input to the RNN model. During the forward propagation of the model, the RNN model processes each time step of the input sequence sequentially. At each time step, the model generates a prediction result based on the current input sequence data with the previous hidden state. And based on the current input and prediction result, and

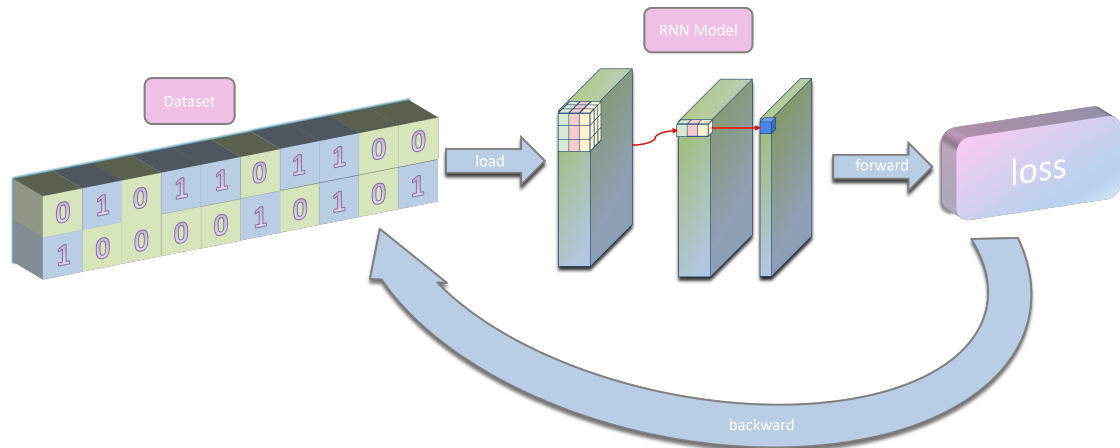


Figure 6: RNN Training: The short sequence data is trained by RNN for prediction and then the loss function is calculated to complete the forward propagation, in order to reduce the loss function the structural parameters in the RNN are adjusted and the data is re-inputted for the updating of the loss function to complete the back propagation.

the previous hidden state, the hidden state is updated. This hidden state will continue to be used in the next time step to help the model better understand the temporal dependencies in the sequence. After the prediction of one time is completed, the RNN model will repeat the prediction process. That is, the prediction result from the previous time step is repeatedly used as the input for the next time step, and then the forward propagation process is performed again. This process will continue until we predict the result of the entire long sequence.

5.3. Prediction Results

The variation of the prediction results with respect to the loss function during training and prediction is shown in Figure 8. The blue curve in the figure 8(a), 8(b), 8(c) is the deformed curve obtained after simulation of the hydrogel model with a length of 15 cm sequence, i.e., the true value, and the yellow curve is the prediction curve consisting of the 300 points used as the prediction result. The training process includes a total of 80 epochs, totaling more than 30,000 iterations of training. As can be seen from the figure 8(d), the loss function decreases rapidly in the early stage of training, indicating that the training results fit the real deformation results quickly, after 10 epochs the decrease of the loss function tends to level off, but is still decreasing steadily, and after all the 80 epochs have been completed, the loss function decreases to $1.7e-7$, and the training curve basically agrees with the real curve, and the RNN model has learned the deformation features of the hydrogel model with a length of 10cm. The results of the prediction of the hydrogel sequence model with a length of 15cm by this model are shown in the yellow curve in the figure 8(a), 8(b), 8(c). As can be seen from the figure 8(a) 8(b) 8(c), the change of the predicted loss function is basically the same as the change rule of the training loss function, in the first 10 epochs when the rapid fit with the real results, and then the decline gradually stabilized, and finally stayed in the vicinity of $2.2e-5$, and there is still a tendency to continue to decline, if we continue to increase the number of epoch, the prediction results can also be

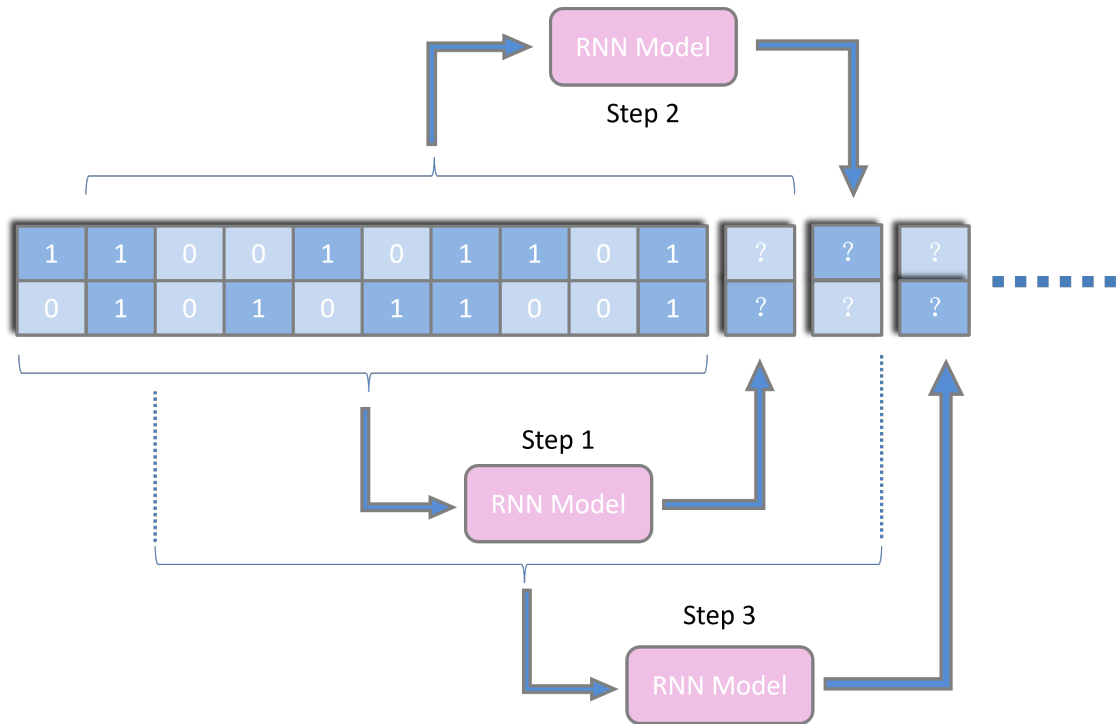


Figure 7: RNN Prediction: Taking the latest 10 time steps as a group, the results of the next time step are predicted from this group based on the data extracted during the training of the RNN model and the grouping of the time steps is updated.

more accurate. Compared with the training loss function, the predicted loss function is $1e2$ higher, but in the comparison chart with the real results, it can be seen that the real deformation trend has been predicted, and to a large extent predicted the specific coordinates of each point. From the comparison of the predicted results with the real results, it can be seen that the RNN model has basically mastered the deformation characteristics of the hydrogel model used in this experiment, and it can use the deformation characteristics of the 10 cm model to predict the deformation of the 15 cm model. The prediction results are highly consistent with the simulation results, and the prediction accuracy can be increased with the increase of training times.

6. Conclusion and Feature

In this paper, we use deep learning to predict the deformation of hydrogel models created by 4D printing technology. We designed a Recurrent Neural Network (RNN) structure and demonstrated the utility of the method to predict the deformation of a long hydrogel based on the deformation parameters of a short model for a two-layered rectangular object created with hydrogel. In general, building and simulating complex models using the finite element method is a time-consuming task. The method submitted in this study can efficiently predict simulation results without finite element analysis, thus significantly reducing time consumption.

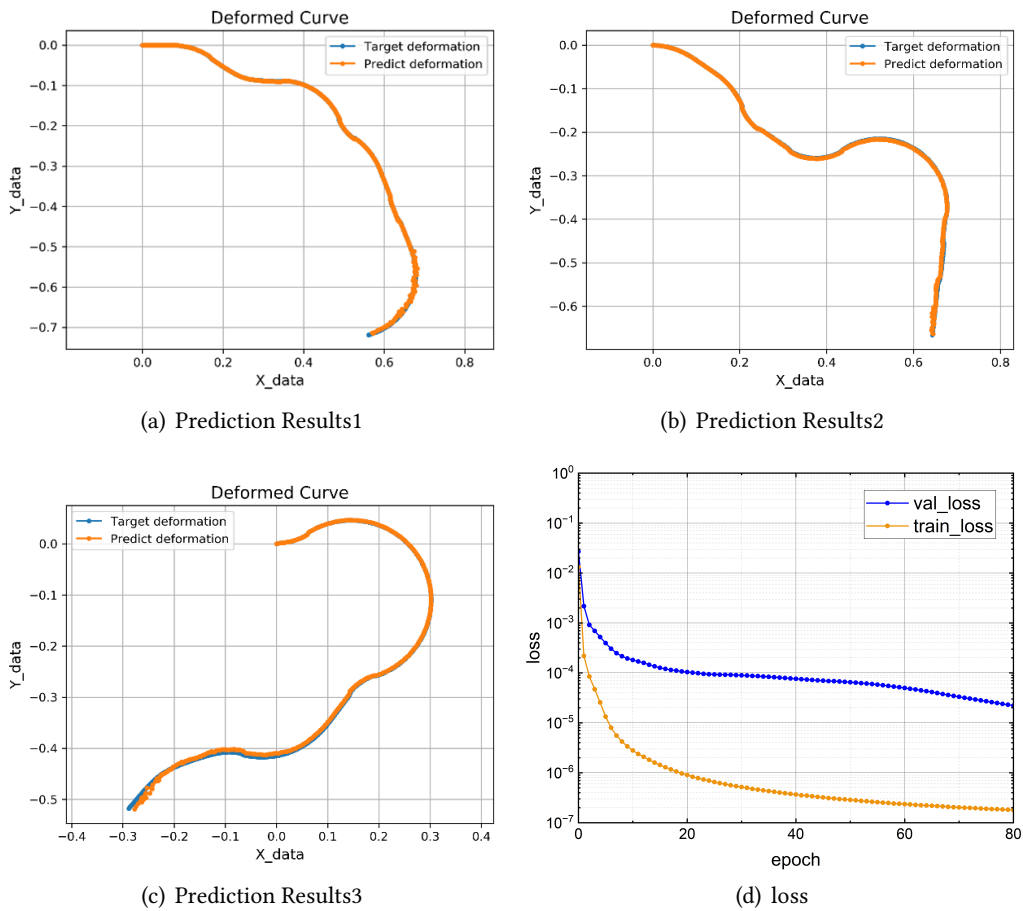


Figure 8: Prediction Results:(a) (b) (c) are three plots of the prediction results in the last epoch, with the x and y axis data being the coordinates of the corresponding axes. (d) is a plot of the variation of the training loss function versus the predicted loss function.

However, at present, this experiment still suffers from the problem that the training phase of the RNN model takes a long time and the prediction accuracy is not high for models with long lengths. In the future, we will continue to improve the neural network structure and optimize the algorithm with new suitable loss function formulas to create RNN networks that allow more accurate predictions and broader applications in the 4D printing domain.

References

- [1] N. Shahrubudin, T. C. Lee, R. Ramlan, An overview on 3d printing technology: Technological, materials, and applications, *Procedia Manufacturing* 35 (2019) 1286–1296.
- [2] X. Kuang, D. J. Roach, J. Wu, C. M. Hamel, Z. Ding, T. Wang, M. L. Dunn, H. J. Qi, Advances in 4d printing: materials and applications, *Advanced Functional Materials* 29 (2019) 1805290.

- [3] S. Ramesh, S. Kiran reddy, C. Usha, N. K. Naulakha, C. Adithyakumar, M. Lohith Kumar Reddy, Advancements in the research of 4d printing-a review, in: IOP Conference Series: Materials Science and Engineering, volume 376, IOP Publishing, 2018, p. 012123.
- [4] F. Momeni, X. Liu, J. Ni, et al., A review of 4d printing, *Materials & design* 122 (2017) 42–79.
- [5] M. Kameoka, Y. Watanabe, M. N. I. Shiblee, M. Kawakami, J. Ogawa, A. Khosla, H. Furukawa, S. Zhang, S. Hirai, Z. Wang, 4d printing of hydrogels controlled by hinge structure and spatially gradient swelling for soft robots, *Machines* 11 (2023) 103.
- [6] M. Champeau, D. A. Heinze, T. N. Viana, E. R. de Souza, A. C. Chinellato, S. Titotto, 4d printing of hydrogels: a review, *Advanced Functional Materials* 30 (2020) 1910606.
- [7] J. Banks, Introduction to simulation, in: Proceedings of the 31st conference on Winter simulation: Simulation—a bridge to the future-Volume 1, 1999, pp. 7–13.
- [8] X. Sun, L. Yu, L. Yue, K. Zhou, F. Demoly, R. R. Zhao, H. J. Qi, Machine learning and sequential subdomain optimization for ultrafast inverse design of 4d-printed active composite structures, *Journal of the Mechanics and Physics of Solids* (2024) 105561.
- [9] X. Yue, H. Li, M. Shimizu, S. Kawamura, L. Meng, Deep learning-based real-time object detection for empty-dish recycling robot, in: 2022 13th Asian Control Conference (ASCC), IEEE, 2022, pp. 2177–2182.
- [10] Z. Li, H. Li, L. Meng, Model compression for deep neural networks: A survey, *Computers* 12 (2023) 60.
- [11] Q. Li, H. Li, L. Meng, Deep learning architecture improvement based on dynamic pruning and layer fusion, *Electronics* 12 (2023) 1208.
- [12] Y. An, M. Wang, L. Chen, Z. Ji, Dcgan-based symmetric encryption end-to-end communication systems, *AEU-International Journal of Electronics and Communications* 154 (2022) 154297.
- [13] Y. Wang, X. Wang, H. Yang, Y. Geng, H. Yu, G. Zheng, L. Liao, Mhagnn: A novel framework for wearable sensor-based human activity recognition combining multi-head attention and graph neural networks, *IEEE Transactions on Instrumentation and Measurement* (2023).
- [14] Y. Yu, X. Si, C. Hu, J. Zhang, A review of recurrent neural networks: Lstm cells and network architectures, *Neural computation* 31 (2019) 1235–1270.
- [15] A. Kag, V. Saligrama, Training recurrent neural networks via forward propagation through time, in: International Conference on Machine Learning, PMLR, 2021, pp. 5189–5200.